HW_2poz3615

Piper Zimmerman

September 9, 2021

2.

a. In this class, I am hoping to learn how to use R more proficiently without having to look the majority of what I'm coding up. This is my first time using R Markdown, so I'm excited to get familiar with that.

```
Objectives<-list("Loops","Data Cleaning","Creating Plots from Loops and Data Analysis")
Objectives
```

```
## [[1]]
## [1] "Loops"
##
## [[2]]
## [1] "Data Cleaning"
##
## [[3]]
## [1] "Creating Plots from Loops and Data Analysis"
```

b.

$$f(x|a,b) = \frac{1}{b-a} \;\; a \leq x \leq b \tag{1}$$

$$f(x|\alpha,\beta) = \frac{1}{\Gamma(\alpha)\beta^{\alpha}} x^{\alpha-1} e^{-\frac{x}{\beta}} \;\; 0 \leq x < \infty, \;\; \alpha,\beta > 0 \tag{2}$$

$$f(x|\beta) = \frac{1}{\beta} e^{\frac{-x}{\beta}} \;\; 0 \leq x < \infty, \;\; \beta > 0 \tag{3}$$

3. (1) For Every Result, Keep Track of How It Was Produced- Make sure you keep track of anything that might influence how you got a result, such as a parameter or input. A challenge in performing this step could be labeling and keeping everything organized so you know which code gave you which results.

   (2) Avoid Manual Data Manipulation Steps- You should rely on the execution of the programs rather than manually modifying data. A challenge in performing this step could be trying to fix something the easy way manually instead of trying to work it out through the program.

   (3) Archive the Exact Versions of All External Programs Used- Always keep the version of the program that you used to create the original result. A challenge in performing this step could be if your computer won't run the version of the program anymore.

   (4) Version Control All Custom Scripts- Archive all of your codes to ensure that you have the one that gave the intended results. A challenge in performing this step could be keeping the scripts organized and easy to find.

   (5) Record All Intermediate Results, When Possible in Standardized Formats- Store all your results on the way as they might be helpful when you are looking back. A challenge in performing this step could be having too many results that they get mixed up and hard to find.

(6) For Analyses That Include Randomness, Note Underlying Random Seeds- If using randomness, the results will vary unless you set the seed the same, so take note of the seed and which steps involve randomness. A challenge in performing this step could be picking out all the steps with randomness. If one is missed, and the seed is not set, it will provide different results.

(7) Always Store Raw Data Behind Plots- Saving raw data behind the plot will reduce time and effort needed to modify the plot in the future. A challenge in performing this step could be keeping the data organized in order to make quick adjustments to the plots.

(8) Generate Hierarchical Analysis Output, Allowing Layers of Increasing Detail to Be Inspected- If you are using summarized results, you should also look at the data that is underlying the summaries. A challenge in performing this step could be if the summarized data doesn't have an option to see the full set.

(9) Connect Textual Statements to Underlying Results- It is important to have more than just the interpretations in text on a paper. The data and results should be easily accessible. A challenge in performing this step could be finding a way to make the data accessible while preserving the professionalism of the paper and not straying from the topic at hand.

(10) Provide Public Access to Scripts, Runs, and Results- Everything should be made public and easily accessible, including the data and source code. A challenge in performing this step could be making everything easily accessible and organized.

4.

```
library(data.table)
library(curl)
```

```
## Using libcurl 7.64.1 with LibreSSL/2.8.3
```

```
covid_raw <- fread("https://opendata.ecdc.europa.eu/covid19/casedistribution/csv")
us <- covid_raw[covid_raw$countriesAndTerritories == 'United_States_of_America',]
us_filtered <- us[us$month %in% c(6:7),]
us_filtered$index <- rev(1:dim(us_filtered)[1])
fit<-lm(`Cumulative_number_for_14_days_of_COVID-19_cases_per_100000`~index, data=us_filtered)
```

ai.

```
library(knitr)
kable(summary(us_filtered))
```

	dateRep	day	month	year	cases	deaths	countriesAndTerritories	geoId	countryterritoryCode	popData2019	continentExp	Cumulative_number_for_14_days_of_COVID-19_cases_per_100000	index
	Length:61	Min. : 1.00	Min. :6.000	Min. :2020	Min. :18665	Min. : 242.0	Length:61	Length:61	Length:61	Min. :329064917	Length:61	Min. : 89.76	Min. : 1
	Class :character	1st Qu.: 8.00	1st Qu.:6.000	1st Qu.:2020	1st Qu.:25240	1st Qu.: 500.0	Class :character	Class :character	Class :character	1st Qu.:329064917	Class :character	1st Qu.: 92.43	1st Qu.:16
	Mode :character	Median :16.00	Median :7.000	Median :2020	Median :45221	Median : 767.0	Mode :character	Mode :character	Mode :character	Median :329064917	Mode :character	Median :150.94	Median :31
	NA	Mean :15.75	Mean :6.508	Mean :2020	Mean :44666	Mean : 791.6	NA	NA	NA	Mean :329064917	NA	Mean :170.16	Mean :31
	NA	3rd Qu.:23	3rd Qu.:7.000	3rd Qu.:2020	3rd Qu.:61796	3rd Qu.: 982.0	NA	NA	NA	3rd Qu.:329064917	NA	3rd Qu.:247.01	3rd Qu.:46

	dateRep	day	month	year	cases	deaths	countries	and	Territories	geoId	countryterritoryCode	popData2019	continent	Cumulative_number_for_14_days_of_COVID-19_cases_per_100000	index
	NA	Max. :31.00	Max. :7.000	Max. :2020	Max. :78427	Max. :2437.0	NA		NA	NA		Max. :329064917	NA	Max. :282.72	Max. :61

There are missing values. We have limited ourselves to 61 time points.

aii.

```
library(stargazer)
```

```
##
## Please cite as:
##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.
##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```
stargazer(fit)
```

```
##
## % Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harva
## % Date and time: Mon, Sep 13, 2021 - 13:43:12
## \begin{table}[!htbp] \centering
##   \caption{}
##   \label{}
## \begin{tabular}{@{\extracolsep{5pt}}lc}
## \\[-1.8ex]\hline
## \hline \\[-1.8ex]
##  & \multicolumn{1}{c}{\textit{Dependent variable:}} \\
## \cline{2-2}
## \\[-1.8ex] & `Cumulative\_number\_for\_14\_days\_of\_COVID-19\_cases\_per\_100000` \\
## \hline \\[-1.8ex]
##  index & 4.107$^{***}$ \\
##   & (0.145) \\
##   & \\
##  Constant & 42.853$^{***}$ \\
##   & (5.165) \\
##   & \\
## \hline \\[-1.8ex]
## Observations & 61 \\
## R$^{2}$ & 0.932 \\
## Adjusted R$^{2}$ & 0.930 \\
## Residual Std. Error & 19.922 (df = 59) \\
## F Statistic & 803.464$^{***}$ (df = 1; 59) \\
## \hline
## \hline \\[-1.8ex]
## \textit{Note:}  & \multicolumn{1}{r}{$^{*}$p$<$0.1; $^{**}$p$<$0.05; $^{***}$p$<$0.01} \\
## \end{tabular}
## \end{table}
```

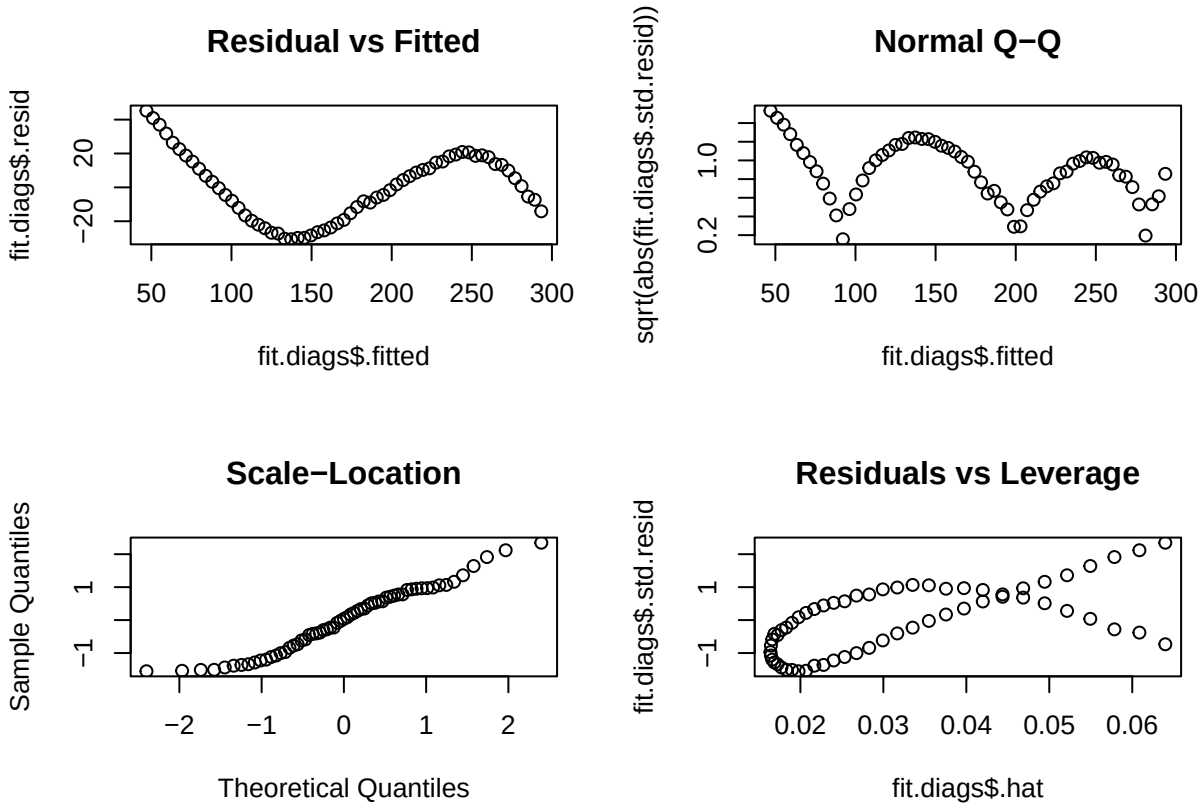
b.

```
library(broom)
fit.diags <- broom::augment(fit)
par(mfrow=c(2,2))
```

Table 2:

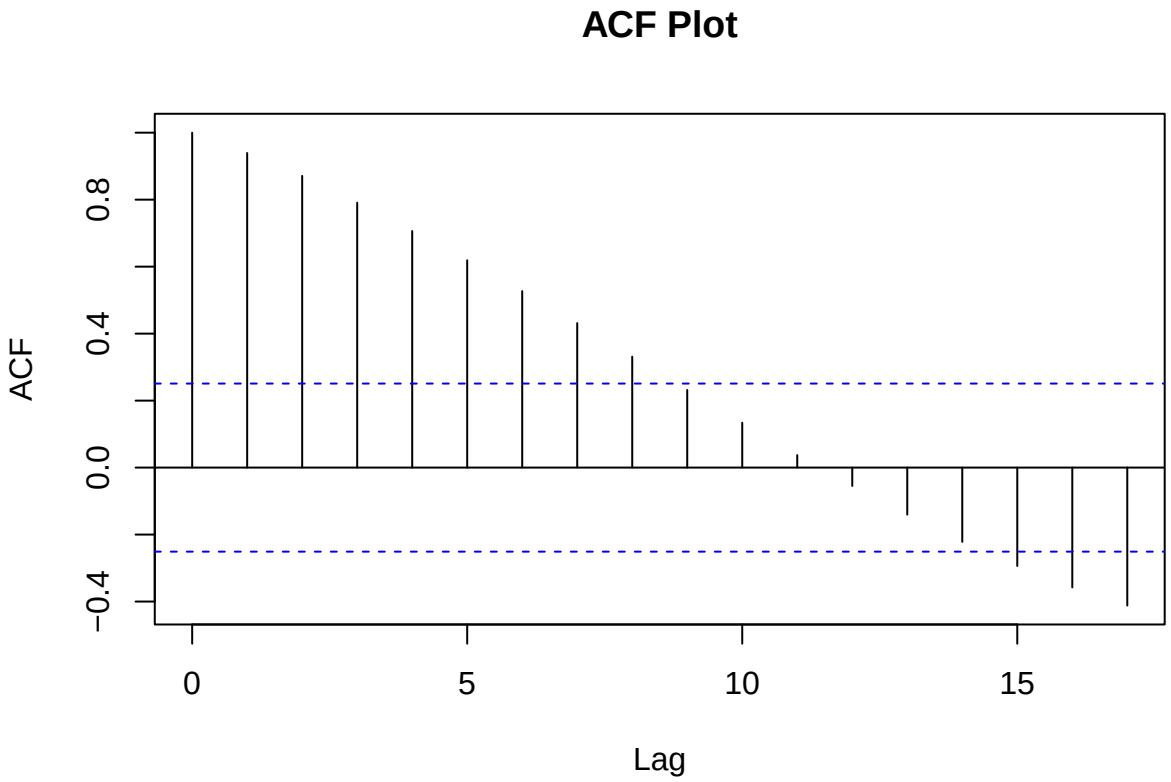
	Dependent variable:
	‘Cumulative_number_for_14_days_of_COVID-19_cases_per_100000‘
index	4.107***
	(0.145)
Constant	42.853***
	(5.165)
Observations	61
R^2	0.932
Adjusted R^2	0.930
Residual Std. Error	19.922 (df = 59)
F Statistic	803.464*** (df = 1; 59)
Note:	*p<0.1; **p<0.05; ***p<0.01

```
plot(fit.diags$.fitted,fit.diags$.resid, main="Residual vs Fitted")
plot(fit.diags$.fitted,sqrt(abs(fit.diags$.std.resid)),main="Normal Q-Q")
qqnorm(fit.diags$.std.resid, main="Scale-Location")
plot(fit.diags$.hat,fit.diags$.std.resid, main="Residuals vs Leverage")
```



c.

```
acf(fit$residuals,main="ACF Plot")
```


ACF Plot
ACF
0.8
0.4
0.0
−0.4
0
5
10
15
Lag